Gene-encoder: A feature selection technique through unsupervised deep learning-based clustering for large gene expression data

Uzma[1], Feras Al-Obeidat[2], Abdallah Tubaishat[2], Babar Shah[2], and Zahid Halim[1]

Abstract

Cancer is a severe condition of uncontrolled cell division that results in a tumor formation that spreads to other tissues of the body. Therefore, the development of new medication and treatment methods for this is in demand. Classification of microarray data plays a vital role in handling such situations. The relevant gene selection is an important step for the classification of microarray data. This work presents gene-encoder, an unsupervised two-stage feature selection technique for the cancer samples' classification. The first stage aggregates three filter methods, namely, Principal Component Analysis (PCA), correlation, and spectral-based feature selection techniques. Next, the Genetic Algorithm (GA) is used, which evaluates the chromosome utilizing the autoencoder-based clustering. The resultant feature subset is used for the classification task. Three classifiers, namely, Support Vector Machine (SVM), *k*-Nearest Neighbors (*k*-NN), and Random Forest (RF) are used in this work to avoid the dependency on any one classifier. Six benchmark gene expression datasets are used for the performance evaluation and a comparisons is made with four state-of-the-art related algorithms. Three set of experiments are carried out to evaluate the proposed method. These experiments are for the evaluation of the selected features based on sample-based clustering, adjusting optimal parameters, and for selecting better performing classifier. The comparison is based on accuracy, recall, false-positive rate, precision, F-measure, and entropy. The obtained results suggest better performance of the current proposal.

Keywords Deep learning, gene expression, clustering, unsupervised learning, genetic algorithm

1 Introduction

Bioinformatics is a domain that merges computing, statistics, and mathematical methods to understanding and solve various biological problems. It mainly includes three sub-disciplines. The first one is to understand the relationship between entities contained in enormous data through the development of novel algorithms and statistical analysis.

✉ Uzma
uzma@giki.edu.pk

Feras Al-Obeidat
feras.Al-Obeidat@zu.ac.ae

Abdallah Tubaishat
abdallah.Tubaishat@zu.ac.ae

Babar Shah
babar.shah@zu.ac.ae

Zahid Halim
zahid.halim@giki.edu.pk

[1]The Machine Intelligence Research Group (MInG), Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan

[2]College of Technological Innovation at Zayed University, Abu Dhabi, UAE

Second is to understand and analyze various kinds of data, such as the deoxyribonucleic acid (DNA) and protein sequences, gene expression, and protein structure. Whereas, third is the efficient access to information through the implementation and development of modern tools. Bioinformatics methods are often utilized for the big data generated through multiple initiatives. Genomics and proteomics are the two important large-scale areas that use bioinformatics methods. Genomics is the study of the organism's genome, which includes the sequences of DNA that determine the entire life of an organism. The genome consists of DNA sequences that include the set of genes carrying the hereditary material from parents to offspring and these transcripts include the ribonucleic acid (RNA) copies. The RNA decodes the genetic information. The analysis and sequencing of the genomic entities which counts both the transcripts and genes in an individual are referred to as genomics. Whereas, the analysis of the complete set of proteins is known as proteomics. Furthermore, bioinformatics is applied in many areas of biology, such as genomics, proteomics, transcriptomics, metabolomics, evolutionary biology, population genetics, precision medicine, and drug design. These areas aim to understand the complex biological system. Devising an unsupervised feature selection technique for the analysis of gene expression data is an open research problem. It selects the feature subset that is more effective for the clustering and classification of various gene expression data.

DNA microarray is used for the measurement of gene expression levels of thousands of genes simultaneously. To figure out the gene function, a subtype of cancer and gene regularity mechanism, biologists measure the expression level in specific experimental conditions [1]. The analysis of gene expression data is getting attention due to its multiple applications in cancer diagnosis, prognosis, and other such domains. The most common analysis of gene expression data is the clustering of cancer samples. The primary aim is to group the samples with similar expression patterns that could help in the discovery of new cancer types. The clustering methods, a subcategory of unsupervised learning, are nowadays being emphasized in the scientific community since after their use in works like [2] and [3]. The clustering methods group the given data points in a way such that the points within a group are more similar to each other than the points in different groups based on all or a set of specified features. Therefore, the clustering analysis divides the gene expression data into groups such that similar genes (or samples) go in one group while dissimilar samples are placed in another group(s). The gene expression data is meaningful for both sample and gene-based clustering. In the gene-based clustering, the co-expressed genes are grouped based on their expression patterns. It treats the genes as objects and the samples as features. However, in sample-based clustering, the samples can be assigned to homogeneous groups. Such clustering treats the samples as an object and genes as features. In gene expression datasets, genes are samples that can be defined numerically as a vector [4] as shown in Eq. (1).

$$O_{i,j} | 1 \leq j \leq f \tag{1}$$

Where, f represents the total number of features and $O_{i,j}$ represents the expression level of the j^{th} feature for i^{th} data observation. The similarity between two objects O_i and O_j is the measure of Euclidean distance. Therefore, the distance between two observations O_i and O_j in an *f*-dimensional space is defined in Eq. (2).

$$Euclidean(O_i, O_j) = \sqrt{\sum_{s=1}^{f} (O_{is} - O_{js})^2} \tag{2}$$

Different clustering algorithms are used for gene expression data such as *k*-means [5], Self-Organization Map (SOM) [6], hierarchical clustering [7], graph-theoretical approaches [8], model-based clustering [9], and Density-based Hierarchical Clustering (DHC) [10] approaches. In the gene expression matrix, there are usually several particular macroscopic phenotypes of samples related to some diseases or drug effects, such as diseased samples, normal samples, or drug-treated samples. The sample-based clustering aims to find the phenotype structure of the samples. The sample's phenotype is discriminated by a small subset of genes referred to as informative genes that are strongly correlated with the class label. The rest of the genes are considered as noise and have no role in the partitioning of interesting samples.

The clustering of samples fall into two main categories: supervised analysis and unsupervised analysis. In a supervised approach, the phenotype information is attached to the samples. These phenotype's information is used for the construction of a "classifier" that contains informative genes. This "classifier" is used for the clustering of samples and predicting the class labels for the incoming samples from the expression profile. Informative genes (i.e., features) are used only for the clustering of the whole set of samples. Due to low dimensional features, usually, *k*-means and SOM are applied for the clustered samples. Unsupervised clustering and informative gene selection is a complex process because of non-availability of the prior knowledge. There are two challenges for the unsupervised clustering that makes it hard to detect the phenotype of clusters and select the informative genes. These are: (1) the gene expression datasets contain a limited number of samples and a huge number of features, therefore, the convolutional

techniques are unable to detect the sample's class properly, and (2) out of all features, only 10% possess the required information [11]. Most of the collected genes are considered as noise because they play no role in the partition of samples. Therefore, it is difficult to choose informative genes for the clustering of samples. There are two methods used to address this problem. The first method reduces the feature's dimensions by the identification of informative genes with the use of some statistical models [12] and then applies the conventional method for sample clustering. Whereas, the second method in an iterative manner use the relationship between the genes and samples for feature selection and clustering the samples simultaneously [13].

The goal of clustering is to partition the objects without having knowledge about their class label. Various clustering approaches have been designed for many application scenarios [14]. These are divided into two main categories, i.e., hierarchical clustering and partition-based clustering. The input to a clustering algorithm is either a pattern matrix or a proximity matrix. In the pattern matrix, each item is represented by a feature vector, whereas, the proximity matrix contains the similarity or dissimilarities between all pairs of points. This work focuses on the partition-based clustering with the pattern matrix as an input. In practice, if more information about the pattern is available, an improved clustering cannot be attained. This is because most of the features have 'noise' which degrades the performance of the clustering process.

The gene expression data contains a lot of irrelevant, redundant, and noisy items. The ratio of informative and noisy data is 1:10 which degrades the performance of clustering if conventional methods are directly applied to the complete feature set. Therefore, the informative feature selection plays a vital role in high dimensional gene expression data for the retrieval of biological information. The feature selection methods are divided into two broad categories. The first category involve supervised, unsupervised, and semi-supervised techniques based on the availability of previous information. The second category contains filter, wrapper, embedded, hybrid, and ensemble methods based on how they concatenate the selection with the model building. All these methods have their advantages and disadvantages. Generally, the hybrid method is better than the wrapper method because it is less prone to overfitting. However, the ensemble method is more robust and flexible [15]. Therefore, the past works used this approach as the feature selection method.

1.1 Problem statement

The significance of gene expression analysis in medical science is high due to the mystery of biological systems. Therefore, one needs to understand gene expression data and extract important information. The analysis of biological data is highly demanding for the biologists to identify various diseases, its types, drug designing, and type of genes, and gene function. The DNA microarray technology can identify the expression level of hundreds of genes simultaneously. The internal view of the gene expression dataset makes it challenging to process**.** The huge dimension of gene expression data contains noise, redundant, and irrelevant items that makes it difficult to analyze. In the literature, the feature selection techniques are used to reduce the dimensions of the data for better gene expression analysis.

Therefore, the proposed framework present a novel unsupervised two stage-feature selection technique for the classification of cancer samples. This framework uses the ensemble of three filter methods using the union aggregation function. The novelty lies in the second stage when unsupervised deep learning is used for the evaluation of each individual of the GA population. The encoder part uses the gene subset represented by the individual as an input. Afterwards, it transforms into a code layer by reducing its dimensions. The decoded part uses the reduced feature subset to generate the original individual feature set. Once the network is trained, *k*-means clustering is applied to the code layer for sample clustering of that specific set of features. Based on this, the problem statement of the present work is as follows.

> *To analyze the gene expression data using ensemble of filters and a GA that evaluates candidate solutions using unsupervised deep learning-based clustering.*

1.2 Key contributions

This work presents a novel approach for the analysis of gene expression datasets. The goal of the current work is to select most informative features for the classification of samples. This method uses samples-based partition-oriented clustering for feature selection. It takes a pattern matrix as an input for clustering. The proposed framework selects the optimum feature subset from high dimensional gene expression data. Therefore, this work is focused on the selection of informative genes for the best clustering of samples. The selection of informative genes to reduce the feature

dimensions in the current framework is based on filter with the wrapper method. Overall, the proposed work makes following contributions.

- Presents an ensemble filter method for feature selection. The ensemble method aggregates the top *n* features recommended by three filter methods, namely, Principle Component Analysis (PCA), the correlation method, and the spectral method.
- Devices an iterative approach to select a set of features and then perform the autoencoder-based clustering and evaluate its validity. This method is a novel, GA-based feature selection technique. The individuals represent the set of selected features. Therefore, an individual (i.e., chromosome) is a binary string where a value of 1 in the gene shows the selected feature. The selected feature set is used as an input to the autoencoder.
- Utilizes the *k*-mean method as a clustering algorithm on the compressed coded layer of the autoencoder. The cluster is validated using the Davies Bouldin Index (DBI). The DBI is used as a fitness value for the individuals. The lower DBI value represents a better set of features. This method gives the best feature set that efficiently clusters the samples.
- Based on the best feature set, the sample classification and also its autoencoding is performed.

The performance of the proposed work is evaluated using both internal and external measures of cluster validity. Whereas, for the evaluation of classifier the confusion metrics is used. The DBI, Dunn index (DI), Silhouette coefficient (SC) are utilized as internal measures. Completeness, homogeneity, v-measure score, normalized mutual information (NMI), accuracy, recall, precision, false-positive rate, f-measure, and entropy are opted as external measures.

The rest of the paper is organized as follows. Section 2 presents the related work. The proposed solution is explained in Section 3. Section 4 lists the conducted experiments and obtained results. Finally, Section 5 concludes this work and lists a few future directions.

2 Related work

This section covers the past works done in the domain of feature selection techniques and the deep/machine learning methods used for overcoming the clustering challenges in the gene expression data. The section is tailed by the limitations of the existing approaches and problem statement.

The DNA microarray technology simultaneously gives the expression level of thousands of genes [16]. It organizes the gene expression data in the form of a matrix where rows and columns represent samples and genes, respectively. The set of values in the row and column represents the gene expression profile. However, each entry shows the expression level of a gene for a given sample. The need for gene expression data analysis is increasing by every passing day. It is a procedure that extracts valuable biological information that helps in finding cure for various human disorders [17]. The analysis of enormous datasets generated by the DNA microarray technology is challenging for the researchers. Therefore, it is imperative to develop a tool in order to analyze and extract biologically meaningful information from some massive gene expression datasets [18]. In this regard, clustering is a useful learning technique, which can be effectively used for the analysis of large volumes of data. It is applied in various fields, for example, data mining [19], image analysis [20], machine learning [21], bioinformatics [23], and pattern recognition [24]. In the clustering technique, the data is partitioned in different groups based on shared characteristics. When clustering is applied to the gene expression data, the related gene expression data are grouped within one cluster and the dissimilar gene expression data is placed into another cluster. The gene expression data are clustered either by samples or genes. Clustering is one of the important techniques for the analysis of gene expression data. It is an unsupervised technique for multivariate data analysis, which puts the observations into groups based on similarity measures. The clustering of gene expression data is useful in understanding gene functions, subtypes of cells, understanding gene regulation, cellular processes, identification of homology, and cellular processes [25]. The internal view of the gene expression dataset makes it challenging to process. There are various number of genes which further makes the gene expression data complicated. Each gene has several conditions that change with time. On top of this, the dataset generated through microarray array technology contains outliers and noise. The proposed framework uses the ensemble filters with a wrapper method for selecting the important features and ignoring irrelevant, noisy and redundant attributes. The filter method determines the variance of each feature by using numerous statistical tests. The high variance feature considered is more important. The selected features' subset comprises of features having larger variance than the threshold or the top high variance features. The wrapper method selects a subset of features based on the classifier performance. Therefore, it finds the optimal features by iteratively selecting a subset of features based on its performance. Several clustering algorithms have been reported in the past to be used for the analysis of gene expression data. Some of them are discussed below.

2.1 Effect of feature selection on learning algorithm

The selection of the best set of features play a vital role in any learning algorithm. The irrelevant features degrade the performance of these algorithms. In feature selection, eliminating unimportant features reduces the data dimensionality. Feature selection for unsupervised learning has been overlooked in the past due to the unavailability of class label, which makes it difficult to select the relevant attributes. Both filter and wrapper methods use the combinatorial search through the space of possible feature subsets. The combinatorial search techniques are used in various feature selection algorithms [26, 27, and 28]. The work in [29] presents a new methodology named as Simultaneous Clustering and Attribute Discrimination (SCAD) that simultaneously perform clustering and feature selection. It learns the different weight sets for the features of each cluster while grouping them. The weight of the features represent their relevance. The cluster of the set of informative features minimizes the objective function. Their experiments indicate comparatively satisfactory performance of their work due to its ability to determine the cluster dependent features. In [30], the authors propose a novel filter method for feature selection known as the Kernel-Based Clustering method for Gene Selection (KBCGS). Their method selects the important features during clustering based on learning the best weights of the genes. They use the kernel method to revel the intrinsic behavior in the data which captures the relationship among the genes. It assigns different weights to each gene, and then the optimal genes are selected by minimizing the objective function value. The performance is investigated by comparing it with six well-known feature selection methods using eight gene expression datasets. Two classifiers, i.e., *k*-NN and SVM are used for classification. The experiments reveals that KBCGS performs better on average.

2.2 Autoencoder-based clustering

Among the various fields of machine learning, deep learning is currently attracting research community attention. In various cases, deep learning performs well than the past works. The term "deep" represents the number of layers through which the data is transformed. Deep learning is successfully adopted in many fields, for instance, image processing, cancer detection, computer vision, and speech recognition. The autoencoder is a type of Artificial Neural Network (ANN) used for unsupervised learning. Following are a few reports in which the autoencoder is used for unsupervised learning.

Song et al. [31] propose a novel deep learning-based graph clustering technique called grapencoder. The grapencoder takes the graph similarity matrix as an input. Then the sparse encoding output of grapencoder is achieved through the greedy layer-wise pre-training process. It first transforms the original graph into a sparse matrix by stack autoencoder, then applies the *k*-means algorithm on the sparse matrix for clustering. Performance of the proposed work is compared with spectral clustering on various graph datasets. The experiments represent that their work performs better from spectral clustering**.** Chen et al. [32] design a methodology for the analysis of high dimensional image data. The new model is based on a hybrid autoencoder, which combines the Stacked AutoEncoder (SAE), Convolutional AutoEncoder (CAE), and Adversarial AutoEncoder (AAE). The hybrid autoencoder combines the advantages of three autoencoders to learn the low feature representation. Afterwards, the *k*-mean algorithm is applied to the output of autoencoder for image clustering. For testing and comparison, the Modified National Institute of Standards and Technology (MNIST) and Canadian Institute For Advanced Research (CIFAR-10) datasets are used on their proposed model. The experiments indicate that their work is better in terms of Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and unsupervised clustering accuracy.

2.3 Evolutionary algorithms for feature reduction

The microarray data faces multiple challenges because of the high dimensional datasets and a small number of samples. Therefore, feature selection plays a vital role in removing irrelevant, redundant, and noisy information for improving the classification problem. In the past literature, evolutionary algorithms have been used to produce high-quality solutions for the optimization problem. Salem et al. [33] proposed a novel methodology for the classification of human cancers. Their proposed framework uses the Information Gain (IG) for feature selection, it then uses the genetic algorithm for feature reduction. Finally, genetic programming is used for classification. Their work is tested for various thresholds. A feature is selected if its IG value is greater than a predefined threshold, otherwise, it is rejected. Their work is compared with six techniques by considering seven microarray datasets.

Ghosh et al. [34] designed a new technique to overcome the challenges of microarray data. Their work is a metaheuristic approach having 2-stages of feature selection. The first method aggregates three filter methods, namely, Relief, chi-square, and symmetrical uncertainty. It takes the union and intersection of the top-*N* ranked features by all three methods. The second stage uses the first stage as an input for the GA to compute results. Their model use three

Table 1 Key features of current work and past contributions

Works	Feature selection	Computational technique	Chromosome evaluation	Classification	No. of datasets
Salem et al. [33]	IG	GA	-	Genetic programming	7
Rani et al. [35]	MI	GA	Classification accuracy	SVM	3
Ayyad et al. [38]	IG		-	Modified *k*-NN	6
Ghosh et al. [34]	Ensemble of 3 filter method	GA	SVM-base classification	MLP, SVM, *k*-NN	5
Uzma et al. (proposed work)	Ensemble of 3 filter methods	GA	Autoencoder-based *k*-means clustering	SVM, *k*-NN, RF	6

classifiers, i.e., *k*-NN, Multi-Layer Perceptron (MLP), and Support Vector Machine (SVM). Performance of their work is tested on five datasets. The experiments indicate better accuracy and a minimum number of features obtained from their work.

Rani et al. [35] design a two-stage feature selection technique for cancer prediction. In the first stage, the mutual information-based feature selection technique is used. In the second stage, the resultant feature subset from the first stage is used as an initial solution for the GA. The feature subset from the two staged method is evaluated using SVM-based classification. Tiwari et al. [36] present a novel optimization technique for feature selection. Their work uses local and global optimization algorithms. Some preprocessing steps are performed instead of randomly generating the initial population for the global optimization algorithm. The local optimization algorithms, Mutual Information Maximization (MIM) and Sequential Backward Search (SBS), are used for removing irrelevant and redundant features. The noise generated by combining the relevant and non-redundant features is removed by applying the global optimization algorithm. The computation time of the global optimization algorithm is reduced through a better stopping criteria.

Key features of the proposed work and past solutions are shown in Table 1. In the past literature, most of the techniques are designed for the supervised feature selection from the gene expression datasets. Mostly, the filter-based feature selection is used for gene expression datasets due to its large-scale information, which is computationally faster. However, in the past literature, the wrapper-based method provides more accurate classification outcomes than the filter-based method [37]. Different filter methods provide dissimilar feature subsets, therefore, selecting the optimal feature subset is a challenging task for unlabeled datasets. The past work use a single filter-based method for feature selection, such as Information Gain (IG) and Mutual Information (MI) [33, 34, 35]. Afterwards, the top-ranked features are used as an initial solution for the GA. However, different filter-based methods provide different feature subsets, so a single filter method does not give an optimal feature subset. Therefore, the proposed framework uses an ensemble of three filter methods. If one filter technique ignores the important feature, there is a possibility that the other selects it. In the previous work, once the feature is selected by using a filter method, it is then optimized by using a GA. The GA evaluates the chromosomes using SVM as a classifier, and accuracy is assigned as fitness value to the chromosome. However, the current work uses the unsupervised method for feature selection. Performance of the traditional clustering-based method is reduced due to the high dimensionality of gene expression data. Therefore, the current proposal uses deep learning model called the autoencoder network, which transforms high dimensions into low and then applies *k*-means algorithm.

3 Proposed solution

This section presents the proposed solution for the feature selection from the gene expression data. The section starts with the preprocessing followed by the clustering of gene expression datasets, the concept of autoencoder, genetic algorithm, and ensemble of unsupervised features selection techniques. The feature slection techniques employed here include PCA, correlation, and spectral methods. Finally, this section provide details of the proposed work's core component.

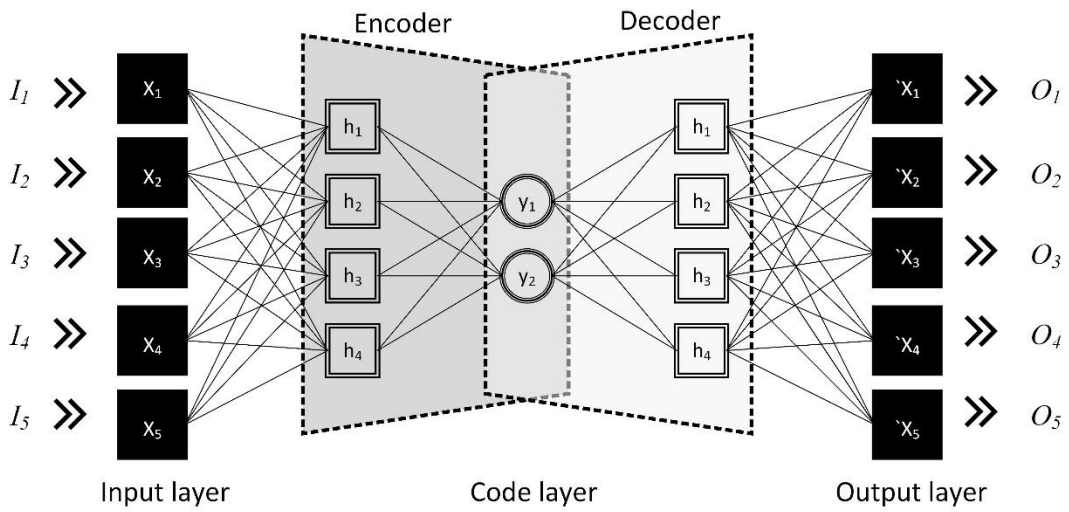

Fig. 1. A sample autoencoder

3.1 Preprocessing

The proposed work use the preprocessing step because the large volume of biological data carries a high level of noise and bias. Therefore, the gene expression datasets require the following one or more preprocessing steps before applying pattern analysis [39].

- The genes expression data exhibit a skewed distribution in which the lower expressed genes are between 0 and 1, while the highly expressed genes are between 1 and infinity. Therefore, when a parametric statistical test is applied to such asymmetric data, it will eventually result in biased results. To overcome this issue, the *log* transformation is used to make the data more symmetric, which is anticipated to give an accurate outcome during statistical tests.
- The replicate handling looks for the repeated gene *ids* in a dataset, which is subsequently replaced by their average value and hence removing the inconsistent repetition.
- The pattern standardization is used that eliminates the scale difference between the features by subtracting the sample average and dividing the value by standard deviation.
- The presence of the missing value of a gene expression is dealt with the average pattern.
- Flat pattern filtering is used that eliminates genes to reduce the complexity of a dataset that is utilized for the biological meaningful analysis.

3.2 Individual components of the proposed solution

The preliminaries used in the proposed framework for feature selection are explained here before going into the details. These include Principle component analysis (PCA), correlation, spectral feature selection, the concept of autoencoder, *k*-means clustering algorithm, and the classification methods.

3.2.1 Principle component analysis

The Principle Component Analysis (PCA) is a linear transformation, which is used to reduce the overfitting problem [40]. It transforms the large quantity of a dependent variable into a small number of independent variables that still comprises a large set of information. Here, the total number (N) of PCA is calculated. Where, N represents a minimum value for a number of samples and associated traits. The PCA_1 is the highest sum squared distance of the projected points from the origin. The following steps are used by PCA-based feature selection.

Step-1: The covariance matrix of size $N * N$ is calculated using Eq. (3).

$$\text{Cov}\,(\text{x},\text{y}) = \sum_{k=1}^{N} \frac{(x_k - \overline{x_k})(x_y - \overline{x_y})}{N-1} \tag{3}$$

Where x_k and $\overline{x_k}$ represent variable and average of the variables, respectively, and N is the total number of variables.
Step-2: The Eigenvalues are calculated using Eq. (4).

$$C - \gamma I = 0 \tag{4}$$

Where, I and γ are the identity matrix and lambda (eigenvalue), respectively, and C is a covariance matrix.
Step-3: The eigenvector for each eigenvalue is calculated.
Step-4: The first two best PCA are selected, i.e., PCA_1 and PCA_2. Then, the formulas given in Eq. (5). and Eq. (6) are applied.

$$x_{vector} = vector_{pca1} * \max\ (T_{features}[pca1]) \tag{5}$$

$$y_{vector} = vector_{pca2} * \max\ (T_{features}[pca2]) \tag{6}$$

Where, $vector_{pca1}$ is the vector for PCA_1 and $T_{features}[pca1]$ is a column of a transform matrix.
Step-5: The most important features are selected by calculating the Euclidean distance between x_{vector} and y_{vector} as shown in Eq. (7).

$$Imp_{features} = sort\left(\sqrt{{x_{vector}}^2 + {y_{vector}}^2}\right) \tag{7}$$

Where, $Imp_{features}$ shows the list of significant features order according to their importance.

3.2.2 Correlation filter measure

The correlation is a statistic that determines the mutual relationship or connection between two or more attributes. It selects the features based on their correlation. Two linearly dependent attributes will have high correlation. Highly correlated features are more linearly dependent and have the same effect on the dependent variable. Therefore, the feature correlation is measured based on a threshold. For instance, if a correlation value is greater than the given threshold, it suggests that the associated features have the same effect and therefore, one or two features are removed. Once the redundant features are excluded from a datasets, the correlation of the remaining features is calculated followed by the sum of the correlation of each feature with variables. Finally, the features are ranked according to their aggregated correlation with the variables in a descending order.

3.2.3 Spectral feature selection

The spectral feature selection method considers feature selection and manifolds learning for reducing a high-dimensional data. In this learning process, the high dimensional data structure is preserved during data conversion into a low dimension [41]. It first constructs the similarity matrix (S) of size $N * N$ from the samples. The similarity matrix data is further used to make a graph. Similarly, the adjacency matrix A, degree matrix D, and Laplacian matrix L of a graph are created. Each feature vector is evaluated by normalized cut and then the features are ranked in a descending order.

3.2.4 Autoencoder

The autoencoder is an artificial neural network used to learn efficient data representation. Aim of the autoencoder is dimensionality reduction by learning a representation for a particular set of data. Autoencoder is used to learn how to encode (reduce) and decode the data back from the reduced data. This kind of representation closely resembles the original representation as much as possible. The autoencoder incorporates the following components: (1) the model learns how to transform the high dimensional data into low dimensional data [42], (2) the code layer contains the compressed representation of input data, (3) the model learns how to reconstruct the data back from the low to high dimension, and (4) the reconstruction loss is a method in which the model measures how close an output is to its input. The model reconstruction loss is minimized by involving backpropagation. The architecture of a simple autoencoder includes the input layer, output layer, and number of hidden layers as shown in Fig. 1.

3.3 *k*-means clustering algorithm

The *k*-means is an unsupervised learning algorithm, which is used for the clustering of unlabeled data iteratively. It categorizes the instances into groups based on their feature similarity. A distance measure is used to assign each data point in one of the groups. It clusters the data points using the following steps: (a) the algorithm initially selects or

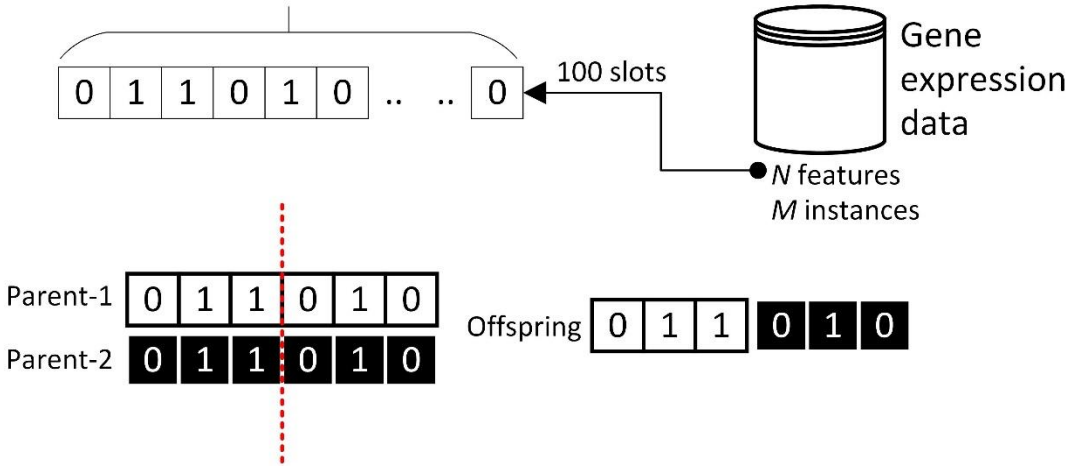

Fig. 2. Chromosome structure and reproduction operations

generates the centroids randomly, (b) it then finds the nearest center for each data point based on the squared distance measure using Eq. (8).

$$\underset{g_i \in G}{dist}\ min(g_i, x)^2 \tag{8}$$

Where, g_i and x represent the group i and data points, respectively. (c) Next, the centers are recomputed by taking the average of all data points contained in a particular center as shown in Eq. (9).

$$g_i = \frac{1}{|N|}\sum_{k=0}^{N} x_i \tag{9}$$

The algorithm repeats steps (b) and (c) until there is no change in the cluster centers or it reaches the maximum number of iteration.

3.4 Classification methods

In machine learning, classification is a supervised problem that recognizes the class of the new observation using a set of known classes. The category is identified based on the training set of data, including observations having the associated classes. The model is to learn from the training set to assign the category to the new observations. The present work utilizes three classifiers for this purpose.

3.4.1 Support Vector Machines (SVMs)

The SVM is a supervised learning model used for classification. SVMs are commonly applied to linearly separable data, however, they can also be used for non-linear classification in a high dimensional feature space. It creates a set of hyperplanes (decision planes) in a high dimensional space to classify the data. The advantage of SVM is its effectiveness in high dimensional space, its versatility (different kernel functions can be used and also the custom kernels), and also memory efficiency. However, if the number of features in larger compared to the number of samples, over fitting may occur.

3.4.2 *k*-Nearest Neighbor (*k*-NN)

The *k*-NN is a method used for both classification and regression. It classifies the data on the basis of majority votes by the *k*-neighbors. For a simple case, if *k*=1, there is one class of the data. The optimal value of *k* can be decided either by inspecting data or a series of experiments. A larger value of *k* is better, as it reduces noise. Votes are decided on the basis of the distance between two points. The distance function can be Euclidean, Manhattan, Minkowski or any other. Distance measures are usually decided on the basis of the type of the data.

3.4.3 Random Forest (RF)

It is a supervised classification method used in machine learning. As the name specifies, it constructs a bag of decision trees somewhat randomly and merges them together to improve the overall result. The concept is called "bagging"

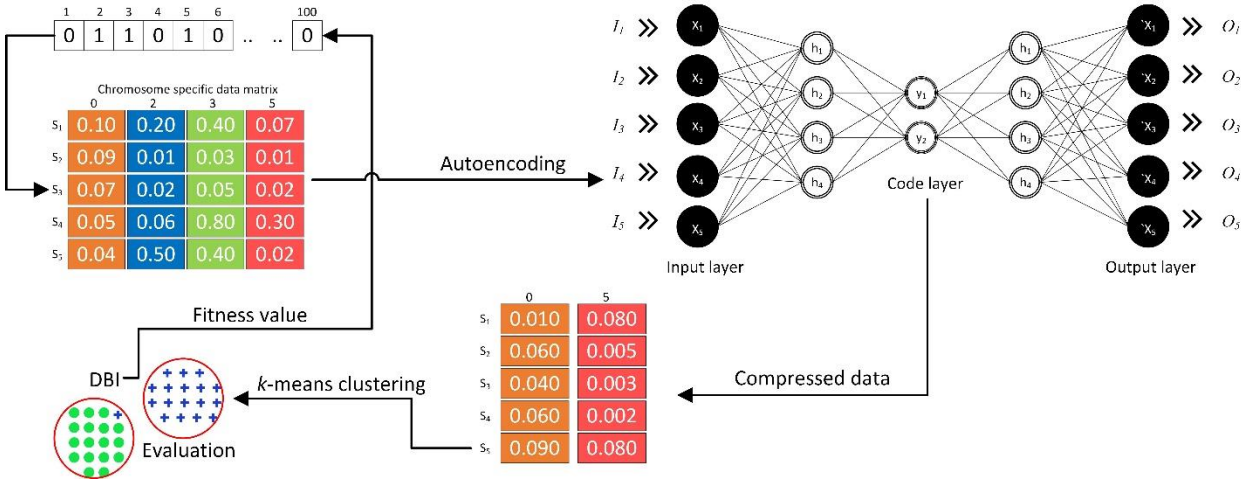

Fig. 3. Chromosome fitness evaluation

method [43]. The RF can be used for classification as well as for regression problems. In classification, RF looks for the random subset of features instead of searching one best feature. It creates the diversity in RF which results in a more stable prediction. Since RF is a set of decision trees termed as ensemble method, it results in better prediction instead of a single decision tree. Ensemble learning algorithms predict on the basis of aggregate decision by multiple predictors.

3.5 The proposed solution core

The proposed solution is a novel two-phase method for the feature selection from gene expression data. It is challenging to select/analyze gene expression datasets having very large number of attributes in contrast to the number of samples. This work implements a two-phase method, which include ensemble filters and wrapper methods for relevant feature selection.

In the first phase, the ensemble filter method is utilized, which reduces the dimensionality of data by removing the irrelevant and redundant features. The ensemble method gives optimal feature set instead of a single filter method because if one method ignores the relevant feature the other one addresses it. Therefore, the aggregation of three filter methods including PCA, correlation, and spectral feature selection is used in this phase. The aggregation method takes into account the union of top-ranked feature subsets. The second phase applies a novel wrapper method on top of the first phase. This method selects a set of features and then evaluates it using autoencoder-based *k*-means clustering.The proposed wrapper method performs two functions: (1) selecting a set of features and (2) clustering the samples based on these sets. This process is repeated several times to choose a set of best features that would eventually preserve more information for clustering. It uses a GA with an autoencoder. Each chromosome describes a set of features. The produced best chromosome of the GA has enough information for classification of samples. The set of optimal features from phase-1 is used as an initial solution. Afterwards, a population of size n is created. Both, *k*-means and autoencoders, are unsupervised learning algorithms. Therefore, the labeled input is not required. The *k*-means algorithm is opted here for the clustering of limited data. The autoencoder is used to determine better representation of the input feature vector. The irrelevant features are removed from the input vector by using the combination of encoder and decoder. In this work, the gene expression dataset is used, having an enormous feature size. The objective of the proposed solution is dimensionality reductions by removing irrelevant and redundant features. Therefore, the chromosome is first given to the autoencoder for its suitable representation. Afterwards, *k*-means clustering is employed on the new representation of the chromosomes. Eventually, the cluster validity index, i.e., DBI is used to assess the chromosome quality.

3.5.1 Problem representation and reproduction operations

The initial solution here consists of binary strings having the size of the original number of features in the data, where 1 and 0 represent the selected and unselected features respectively by using phase-1, i.e., ensemble filter. Selection is the process that picks the chromosomes for later breeding. Truncation selection is used here that orders the population P according to their fitness value. Then, the top $P/2$ fittest chromosomes are selected for breeding. In the current

```
Input: Gene expression dataset
Output: Clusters and classification of gene expression datasets
```

```
1.  M← Dataset
2.  Preprocessing of dataset
3.  Ensemble PCA, Correlation, and Spectral base feature selection techniques with Union
4.  Union(PCA, Correlation, Spectral)
5.  S← optimal features subset
6.  Run GA with Auto encoder
7.      for k=0 to N-generation do
8.      set initial solution of binary string where 1 is set according to S
9.          initial solution ← S
10.         Create population of size m
11. Calculate the fitness value of each chromosome
12. Select the top N best chromosomes
                    Best_chrom ← population
13. Apply genetic operator crossover
                    offsprings ← crossover(Best_chrom)
14. Apply genetic operator mutation
15.       Mutate_ offsprings ← Mutation(offsprings)
16. M_c,M_m ← Dataset is selected according to Best_chrom &
          Mutate_Best_chrom
17. Run Autoencoder on M_c & M_m for compressed data
18. Apply K-means on(code layer)for clustering
                Clusters ← K-means[compressed-data]
               FitnessM_c, FitnessM_m ← DBI (Evaluate clusters)
19. Complete population from selected parents/offsprings and sort them according to fitness
           Best_chrom =sort (FitnessM_c, FitnessM_m)
    Repeat step 14 to 20 until termination criteria is not meet.
20. End
21. Select the top best chromosome from GA
           Chromosome=Best (fitness value)
22.        Run autoencoder for the selected Chromosome and then Apply
23.        k-means based clustering.
24.        Use step 22 for classification.
25. Evaluate step 23 by using internal and external measures
26. Evaluate step 24 by using confusion matrix
```

Algorithm-1. Complete pseudocode of the proposed solution

solution, one point crossover procedure is adopted. It selects the mid of the chromosome as a crossover point, and then the tails of two parents are swapped to generate the offspring. The swap mutation is a genetic operator, which has been implemented in the current work. The swap mutation randomly selects any two genes of an individual and then swaps their locations. The mutation rate represents the percentage of genes that are swapped in a chromosome. Fig. 2 demonstrates the chromosome structure, crossover, and mutation procedures adopted in the present work.

3.5.2 Autoencoder-based *k*-means clustering

The proposed framework first utilizes an autoencoder for the preprocessing of a feature subset, followed by a *k*-means algorithm to evaluate the selected features. The architecture of the autoencoder which has been used for each feature subset (individual) is explained in the following. The autoencoder takes the individual chromosome as an input and the number of neurons in the hidden layer is defined using Eq. (10).

$$h_n = \frac{(P_n)}{2^l} \tag{10}$$

Where, h_n and P_n are number of neurons in the hidden layer and previous layer, respectively. In the given equation, l represent an even number.

The candidate solution is a binary string where 1 as an allele represents the selected feature. Therefore, the number of 1's contained in an individual represent the input for the autoencoder. If x_i is a vector of features having length N, the weights W_e , W_d, and biases b_e and b_d and output $\hat{x}_i$, then the mapping relationships are defined in Eq. (11).

$$c_i = x_i W_e + b_e \tag{11}$$

$$\hat{x}_i = W_d c_i + b_d \tag{12}$$

Where, c_i is the compressed form of x_i and $\hat{x}_i$ is the reconstructed form of x_i. The mapping from x_i to c_i is used as a preprocessing step for clustering. This model learns the features from unsupervised data. Therefore, the detected features are efficient for clustering the unsupervised data. When the autoencoder is trained by adjusting the weights

and minimizing the error, the *k*-means method is applied to the compressed data for the purpose of clustering. Different parameter setting used in this work are listed in Table 3.

3.5.3 Fitness function

The present work use autoencoder-based *k*-means clustering to form groups and the fitness function to evaluate the clustering quality is DBI. Each chromosome represents a set of features. Hence, a particular chromosome is evaluated by applying the autoencoder on a set of features. Afterwards, the *k*-means clustering is used on the compressed data

Table 2 Parameter settings

Parameters	Values
Batch size	20
Epochs	10
Generation	100
Population size	40
TopN	100
Mutation rate	80%
Fitness function	DBI
Reproduction operations	One point crossover & swap mutations

Table 3 Descriptions of the datasets

Datasets	Number of samples	Number of genes	Classes
Leukemia	72	3571	2
DLBCL	77	7070	2
Colon cancer	62	2000	2
Lung cancer	181	12533	2
Prostate cancer	136	12600	2
Center nerves system	61	7129	2

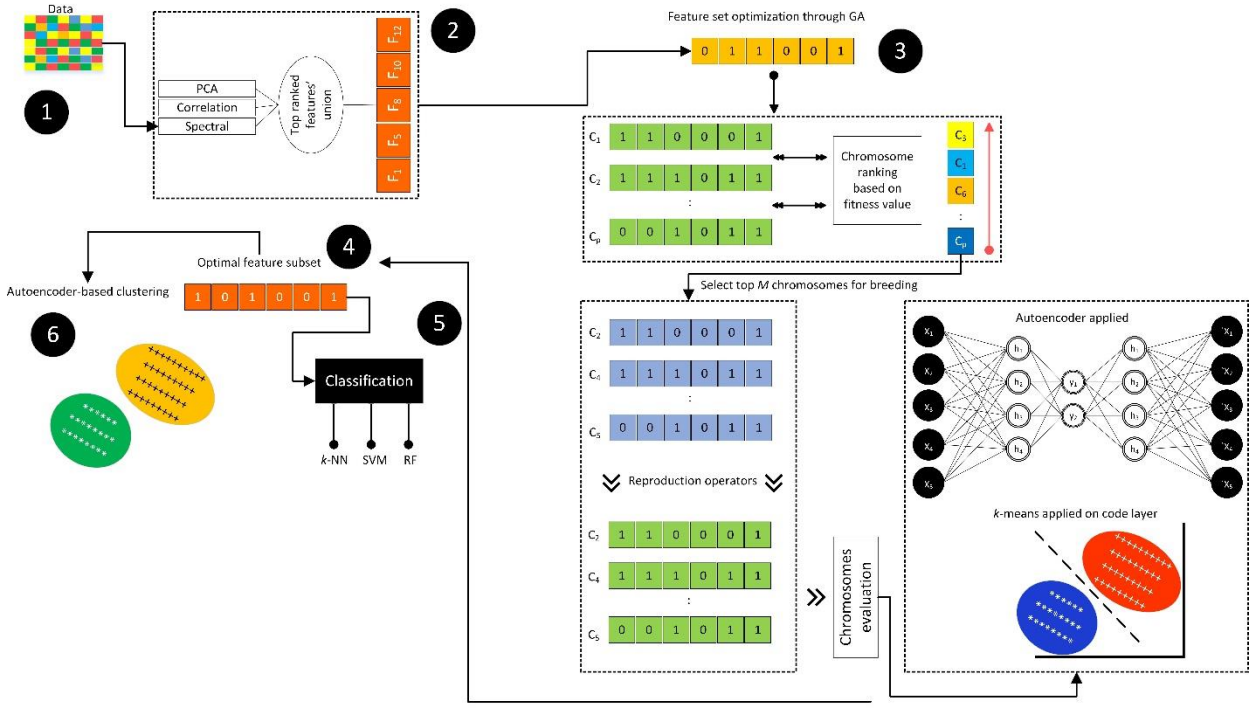

Fig. 4. Overall working of the proposed solution

produced via autoencoder. The generated clusters are then evaluated using DBI. The maximum DBI value show better set of features. The evaluation of fitness value is shown in Fig. 3.

Algorithm-1shows the complete pseudocode of the proposed solution. The focus of the proposed framework is an optimal feature selection technique for the analysis of gene expression data. The proposed framework performs this task in six board phases. In the first phase, the proposed gene encoder takes the gene expression data as input. These are cancer datasets having *S* samples and *F* features. In the second phase three filters methods are applied on the data. After that, the top-ranked features are grouped with the help of union aggregation. This step generates a set of features. In the third phase, a GA is applied for optimization, where the initial solution is a subset of features generated from second phase. The GA population is created based on directed transpositions. The fitness values are calculated and assigned to the set of chromosomes. The chromosomes are then ordered based on their fitness value. Afterwards, top N/2 chromosomes are chosen for breeding. Each mutated chromosome sample is clustered through unsupervised deep learning-based *k*-means algorithm. The validity index, DBI, is assigned to the generated clusters which is used as a fitness value. The selected and mutated chromosomes are then ranked according to their fitness value. Next, this new population is used for crossover and mutation. The whole process is repeated for a fixed number of iterations. In the fourth phase top chromosomes are chosen that represent the best feature subset. This set of features represent the optimum features subset. In the fifth step the generated optimal features set via GA is used for cancer prediction. For this, three classifiers are utilized. In the final stage, the selected optimal set of features are also used for sample-based clustering. The proposed model can be applied for both clustering and classification of gene expression data. Fig. 4 visually demonstrates the complete working of the proposed solution.

4. Experiments and results

This section presents the conducted experiments for the evaluation of the proposed work. For this, six benchmark gene expression datasets are utilized. The proposed idea is compared with four state-of-the-art algorithms, i.e., Salem et al. [33], Ghosh et al. [34], Rani et al. [35], and Modified *k*-Nearest Neighbor (MKNN) [38]. Three types of experiments are carried out for assessing the framework (a) evaluation of the selected features using sample-based clustering (b) analysis of parameter setting, i.e., Top*N* and mutation rate, and (c) evaluation of three classifiers, i.e., SVM, *k*-NN, and RF. The parameter setting mentioned in Table 2 is used in all experiments.

4.1 Datasets

For comparing the present work with the state-of-the-art methods, six benchmark gene expression datasets are utilized, namely, leukemia, DLBCL, colon cancer, lung cancer, and center nerves system [38]. These gene expression datasets contain fewer samples and a large number of genes. The information of these datasets is shown in Table 3. These datasets have two class labels and high dimensional features (i.e., genes). The CNS, colon, and prostate cancer describe the normal and cancerous samples. Whereas, the remaining datasets are about the samples of various types of cancer. The leukemia datasets include 3571 features and 72 samples. The number of samples belonging to ALL of type cancer is 47, and 25 belongs to type AML. The DLBCL dataset has 77 and 7070 number of samples and features, respectively. There are 58 samples having DLBCL type cancer, and 19 samples have class label FL. The colon dataset has 2000 features. Where, the number of samples is 62 having 22 healthy instances and 40 are cancerous samples. The largest dataset is the prostate cancer having 12600 features and 136 samples. Where, the cancerous samples are 77 and 59 are normal. The CNS datasets has 7129 features. The number of samples in this dataset has are 61.

4.2 Performance metrics

The proposed solution utilizes the unsupervised feature selection techniques for the analysis of gene expression data. The tasks of cancerous sample prediction and the clustering of gene expression data are performed based on the selected feature subset. It aims to select a relevant feature subset based on some criteria (their expression level) from the original feature subset. The feature selection is used for many reasons, such as removing redundant and irrelevant features, dimensionality reductions, improving the performance of the learning model, and reducing the amount of data needed for learning. The effectiveness of the classification and clustering problem depends on the selected features' relevance. Confusion metric is used here to evaluate the performance of the classifier. The clustering of gene expression data is employed based on the chosen feature subset. Clustering is an unsupervised method, hence no information about the class is provided. Quality of a clustering algorithm is gauged using its results. For this, a number of cluster validation technique are used for finding the goodness of clustering algorithms. The proposed framework is evaluated using 10 performance metrics. These include both internal and external validation measures [45].

4.2.1 Internal validation measures

Internal validation measures rely only on information available in data without any external evidence (i.e., the class label). The internal validation measures aim to select the optimal number of clusters and the best clustering method [46]. This measure evaluates the compactness and separation. Compactness measures how closely related are the cluster elements. A high compactness value represents low variance between objects of a cluster. Whereas, the separation metric represents the heterogeneity between clusters. High diversity depicts well-separated clusters. Following are the internal measures used here as a cluster validity index.

Davies-Bouldin index (DBI): This index define the average similarity of each cluster with its similar cluster. This approach defines that no cluster is related to others. Therefore, the better clustering scheme minimizes the DBI [47]. However, the similarity is a ratio of intra-cluster distance and inter-cluster distance as shown in Eq. (13).

$$DB = \frac{1}{G}\sum_{c=1}^{G} \quad R_{i\neq j}\frac{d(x_i)+d(x_j)}{d(G_i,G_j)} \tag{13}$$

Where, the number of clusters is shown by G. The clusters are labeled as i and j. Where, the $d(x_i)$ and $d(x_j)$ specify the sample distance in cluster i and j, respectively. The $d(G_i, G_j)$ represents the distance between centers.

Dunn index (DI): The DI measure identifies the set of clusters that are well separated and have low variance between its members [48]. The higher DI indicates better clustering. The DI is mathematically defined using Eq. (14).

$$DI = min_{1\leq i\leq c}\left\{ min\left\{ \frac{d\left(G_i, G_j\right)}{max_{1\leq k\leq c}\left(d(x_k)\right)} \right\} \right\} \tag{14}$$

Where, the inter-cluster distance between clusters G_i and G_j is $d\left(G_i, G_j\right)$ and $d(x_k)$ is the distance between cluster members. The parameter c represents the number of clusters.

Silhouette Coefficient (SC): The SC evaluates the object similarity with its cluster as compared to other clusters as shown in Eq. (15). The range of SC value is between 1 and -1. High SC value represents that the object is categorized correctly [49]. Clustering is appropriate if most of the objects have a high value. For the number of objects, if SC values are negative, this represents that the number of clusters is too low.

$$c(i) = \frac{1}{|S_i|-1}\sum_{j\in S_i, i\neq j} d(i,j) \tag{15}$$

Where, c_i represents the average distance between object i and all members of a cluster. S_i represents the number of samples in the cluster i. The $d(i,j)$ show the distance between object i and j in the cluster.

$$o(i) = min_{k\neq i}\frac{1}{|S_k|}\sum_{j\in S_k} d(i,j) \tag{16}$$

Where, $o(i)$ represents the average distance of an object i from another object j of cluster *k*.

$$SC(i) = \frac{o(i)-c(i)}{max\{c(i), o(i)\}} \quad if\ |S_i| > 1$$
$$S_c(i) = 0 \qquad if\ |S_i| = 1 \tag{17}$$

4.2.2 External validation measures

The external validation measures are used to evaluate the clustering algorithms base on the ground truth. If the obtained result is similar to the prior knowledge, the solution is considered as a good clustering algorithm. The proposed framework is also evaluated using seven external measures. These are listed in the following.

Normalized Mutual Information (NMI): The NMI is calculated based on the ground truth and the predicted class label as shown in Eq. (18). A higher NMI score represents better clustering formation [50].

$$NMI(Y_{label}, C_{label}) = \frac{2*I(Y,C)}{[H(Y)+H(C)]} \tag{18}$$

Where, $I(Y,C)$ represents the mutual information between the class and cluster label as defined in Eq. (19).

$$I(Y,C) = H(Y) - H(Y|C) \tag{19}$$

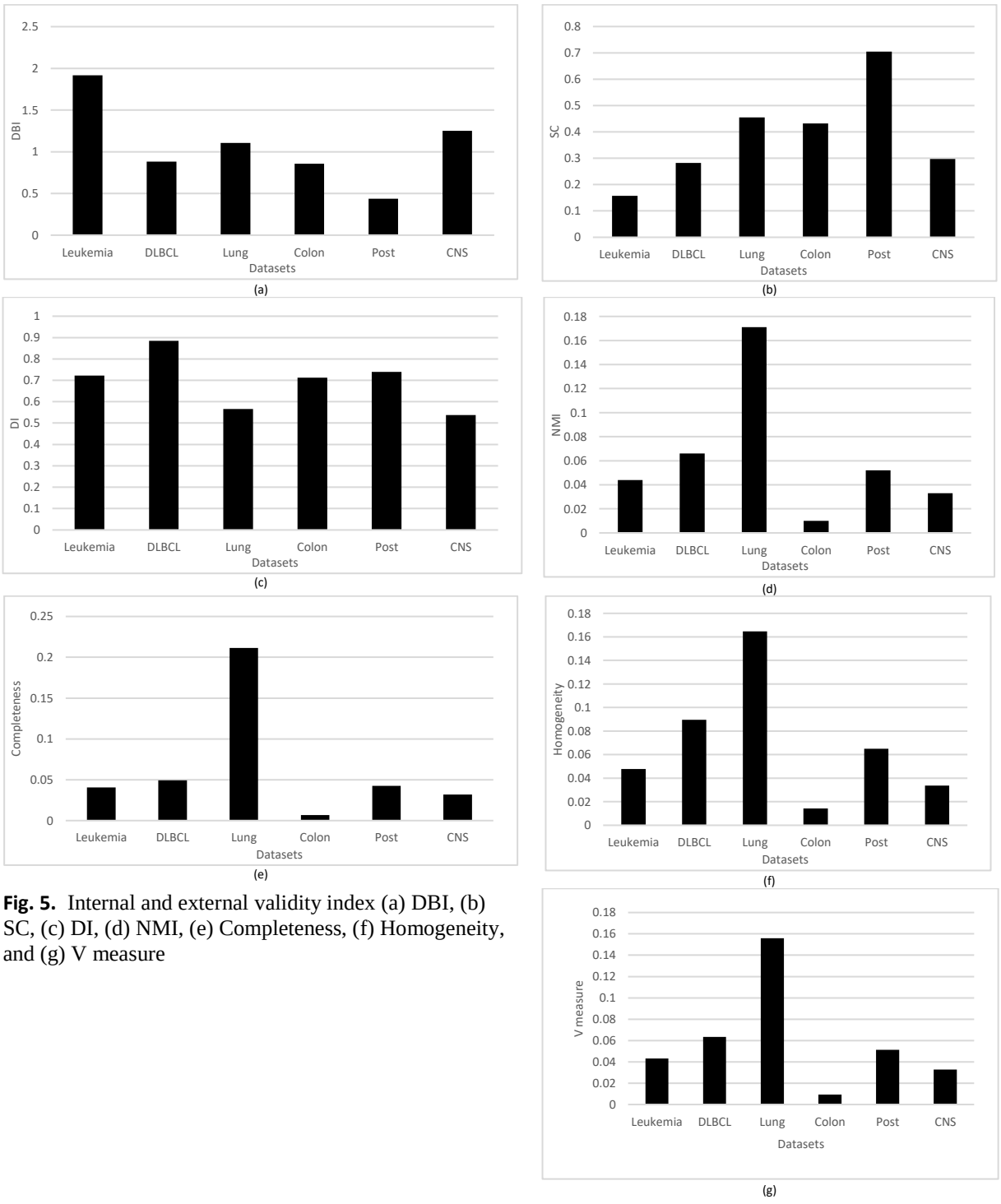

Fig. 5. Internal and external validity index (a) DBI, (b) SC, (c) DI, (d) NMI, (e) Completeness, (f) Homogeneity, and (g) V measure

The entropy of the class label is represented by $H(Y)$. However, $H(Y|C)$ is the entropy of resultant cluster labels.

Entropy: The entropy measure represents purity of a cluster. A value of 0 for entropy indicate that the objects contained in all clusters have a single class label. The higher values of entropy means that the members of a class have distinct class labels [51]. For each cluster, the label's distribution is defined in Eq. (20).

$$E_{ci} = \sum_i p_{ij}\, log\,(p_{ij}) \tag{20}$$

The entropy for a dataset is computed by adding all clusters' entropy and is defined in Eq. (21).

$$entropy = \sum_{i=1}^{c} \frac{n_{ci}}{n} E_{ci} \quad (21)$$

Where, c denotes the number of clusters, n_{ci} is the size of the cluster, and n is the number of points.

F-measure: F-measure is also called the balanced F-score or F-score. It is a harmonic mean of recall and precision [52]. First, the recall and precision of a cluster for each class in computed using Eq. (22) and Eq. (23).

$$Recall_{(li,cj)} = \frac{s_{li,cj}}{t_{li}} \quad (22)$$

$$Precision_{(li,cj)} = \frac{s_{li,cj}}{t_{cj}} \quad (23)$$

Where, $s_{li,cj}$ is the number of samples belonging to label j, t_{li} is the total number of samples belonging to label i, and t_{cj} is the number of samples in cluster j. The F-score of cluster and class is defined in Eq. (24). The F-measure values are in the range of 0 and 1, the higher value represents better clustering.

$$F_{(i,j)} = \frac{2Recall_{(li,cj)}Precision_{(li,cj)}}{Precision_{(li,cj)}+Recall_{(li,cj)}} \quad (24)$$

Completeness, Homogeneity and V-measure score: The completeness criteria for the clustering algorithm states that all samples belonging to the label i are assigned to the same cluster. Where, the homogeneity of a clustering algorithm means that all members of a cluster belong to the same class label [53].

$$Completness_{score} = 1 - \frac{H(l|c)}{H(l)} \quad (25)$$

$$Homogeneity_{score} = 1 - \frac{H(C|l)}{H(c)} \quad (26)$$

Where, $H\ (C|l)$ represent the conditional entropy of a class given its cluster as shown in Eq. (27).

$$H(l|C) = -\sum_{l=1}^{|L|} \sum_{c=1}^{|C|} \frac{S_{l,c}}{n_c} . \log\left(\frac{S_{l,c}}{n_c}\right) \quad (27)$$

and $H(l)$ is the entropy of the class

$$H(l) = -\sum_{l=1}^{|l|} \frac{n_l}{n} . \log\left(\frac{n_l}{n}\right) \quad (28)$$

Where, $|C|$ and $|l|$ represent the number of clusters and classes respectively, n_l and n_c show the number of samples belonging to the class and cluster, respectively. The parameter $S_{l,c}$ is the number of samples belonging to class i assigned to cluster c and n is the total number of samples. The score ranges between 0 and 1. Higher score represents better result. The V-measure is the harmonic mean of completeness, and homogeneity defined in Eq. (29).

$$v = 2\frac{\text{homogeneity.completeness}}{\text{homogeneity+completeness}} \quad (29)$$

Accuracy: Accuracy is a ratio of correctly predicted positive observation and all observation as defined in Eq. (30). The high accuracy shows a better classification model [54]

$$Accuracy = \frac{True_{positive}+True_{negative}}{True_{positive}+False_{positive}+False_{negative}+True_{negative}} \quad (30)$$

False Positive Rate (FPR): The False positive rate is calculated by using Eq. (31)

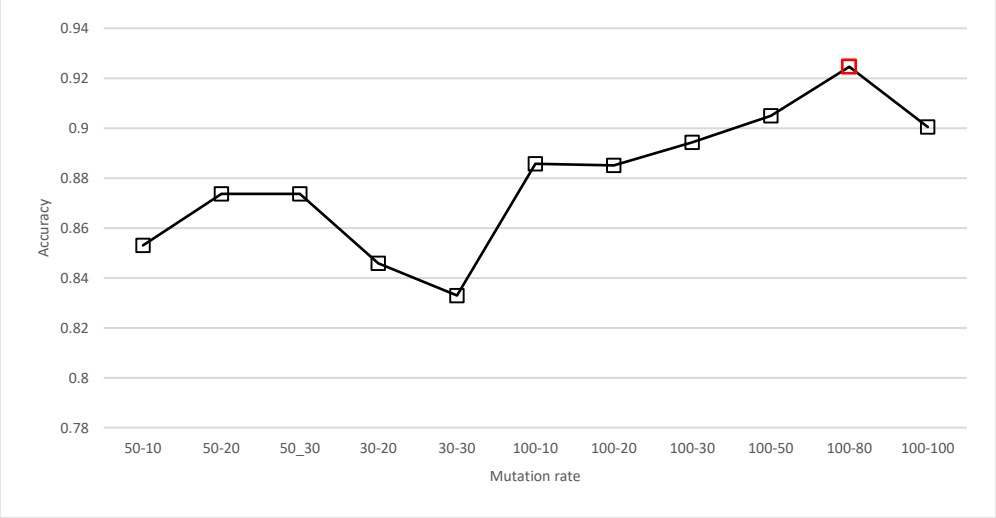

Fig. 6. Results for selecting value of *TopN* and mutation rate

$$FPR = \frac{False_{positive}}{False_{positive}+True_{negatiev}} \quad (31)$$

Where, $False_{positive}$ is the number of positive observations that are predicted as negative. The $True_{negatiev}$ is the number of negative observations that are also predicted negative. The range of this measure is between 1 and 0.

Precision: Precision, also known as Positive Predictive Value (PPV), is defined as the ratio of total correct positive predictions and all positive predictions. The best value of specificity is 1 and its worst value is 0. Eq. (32) shows the computation of precision.

$$Precision = \frac{TP}{TP+FP} \quad (32)$$

Recall: The recall is a ratio of correctly predicted positive observation and all observation of the actual class. The recall is defined in Eq. (33)

$$Recall = \frac{True_{positive}}{True_{poositive}+false_{negative}} \quad (33)$$

4.3 Experiments for evaluating the selected feature based on clustering

The proposed work uses cluster-based feature selection technique. Therefore, this experiment aims to use the internal and external validity index to evaluate the clusters generated for the selected feature subset. An experiment is performed on how well is the clustering formation generated for the selected feature subset. This indirectly represent the relevant feature subset selection. This experiment uses the *k*-means clustering to group the samples for the selected feature subset. The DBI, DI, and SCI are the internal cluster validity indices and NMI, completeness, homogeneity, V-measure score are used as external measures to evaluate the resultant clusters for a selected feature subset. Fig. 5 represents the internal and external validity indices for the clustering of samples with respect to the selected feature subset. As shown in Fig. 5, the SC value for the largest dataset, i.e., prostate cancer, is 0.704833. For the second-largest dataset, Lung-cancer, it is 0.454905. These datasets are largest and smallest respectively in terms of the number of samples and features. However, for CNS and DLBCL datasets, SC values are 0.297281 and 0.282616, respectively. Where the CNS dataset is larger than DLBCL in terms of features count. The SC value for colon-cancer is 0.43193, which is better than the ones obtained for DLBCL and CNS. This is because its feature dimension size is smaller than the other datasets. For the smallest dataset, i.e., leukemia, SC value is 0.156963. This suggest that the proposed solution works well for datasets having a large number of features and smaller samples. The average DI value for all datasets is 0.693567. For the largest and smallest dataset, its value is 0.7225 and 0.7225, respectively. Only for CNS data, its value is a bit worse. This is because it has the smallest sample size as compared to the other five datasets. The DBI value for all datasets show that its average value is 1.0757. Its value for the prostate cancer and leukemia datasets is 0.4381 and 1.9175, respectively. Similarly, the obtained NMI, completeness, and homogeneity values suggest that the proposed work achieve the goal of performing well for datasets having less number of samples than the number of features.

Table 4 Accuracy of proposed work for various values of *k* for the *k*-NN classifier

	Datasets						
	Leukemia	DLBCL	Lung	Colon cancer	Prostate cancer	Center nerves center	Average
k=3	0.66667	0.86250	0.82857	0.76923	0.86486	0.86486	0.80945
k=5	0.74667	0.87500	0.80714	0.76923	0.85946	0.69231	0.79163
k=7	0.70667	0.87500	0.80714	0.76923	0.85946	0.69231	0.78497

Table 5 Performance of the proposed framework with RF classifier

Dataset	Accuracy	Recall	False positive rate	Precision	F measure	Entropy
Leukemia	0.6962	0.5000	0.0417	0.7500	0.7855	0.0000
DLBCL	0.7725	0.4933	0.0727	0.7833	0.5965	0.0000
Lung	0.6688	0.4667	0.2045	0.5500	0.5000	0.0000
Colon cancer	0.8077	1.0000	0.2083	0.3208	0.4750	0.0805
Prostate cancer	0.9459	0.9667	0.1429	0.9669	0.9663	0.0323
Center nerves center	0.9730	0.9667	0.0000	1.0000	0.9831	0.0000

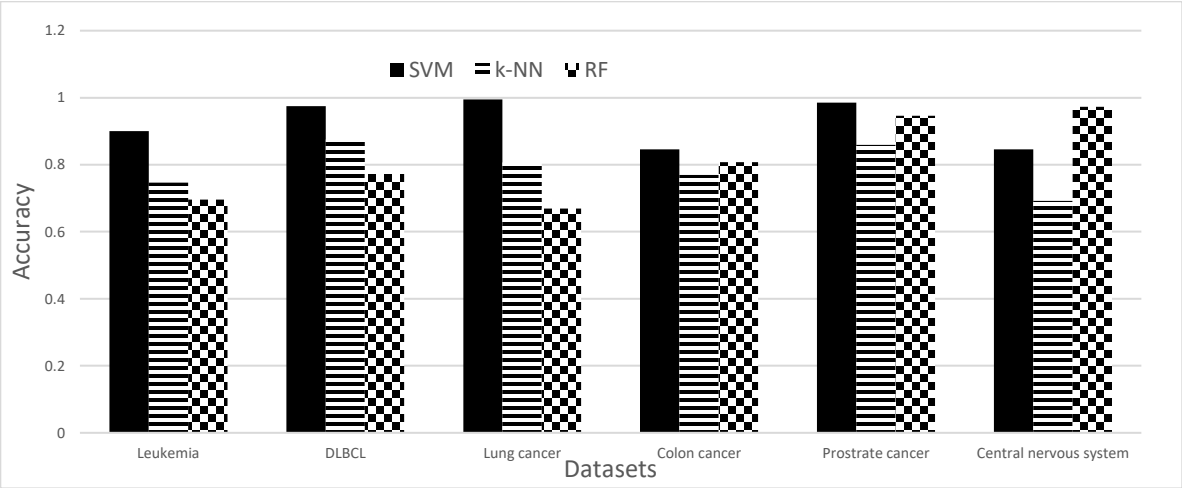

Fig. 7. Accuracy of the proposed framework on three classifiers

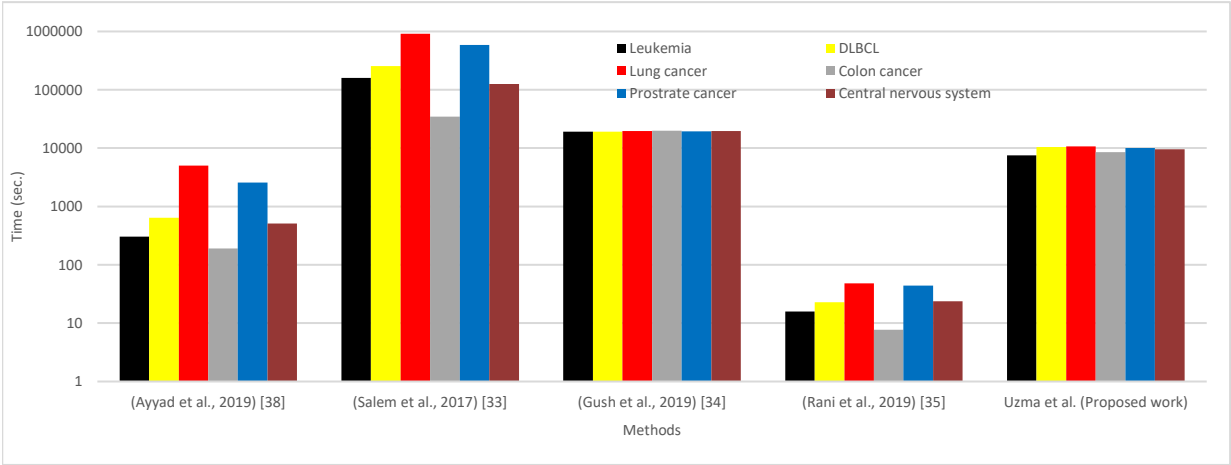

Fig. 8. Performance comparison based on computation

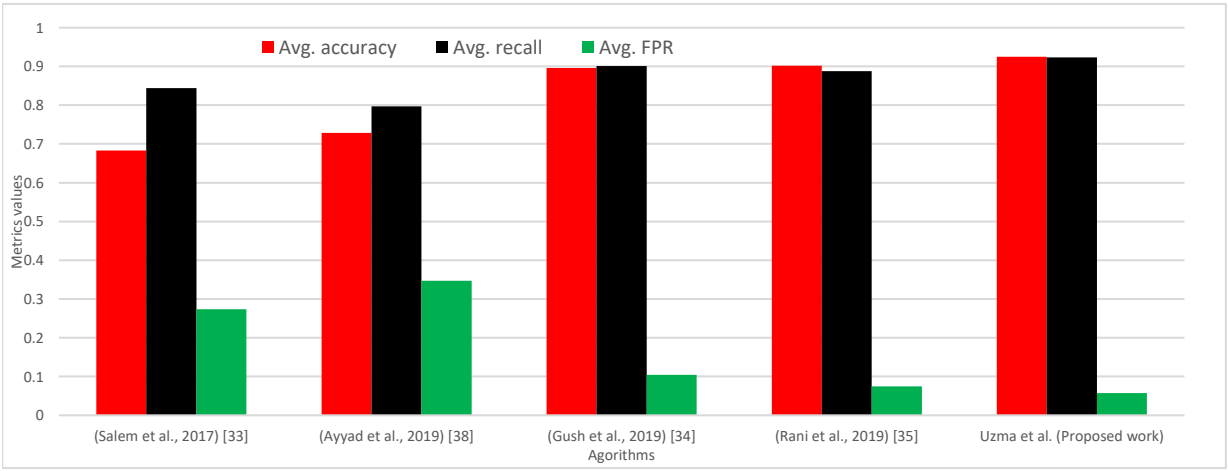

Fig. 9. Average accuracy on six datasets for the five competing methods

4.4 Experiment for setting parameters

This experiment is conducted for setting the two key parameters, i.e., Top*N* and mutation rate. The Top*N* is the highly ranked *N* features that are selected by using the filter method. The mutation rate is the number of genes that are swapped for each individual. The aim of this experiment is to figure out the best parameters for the proposed work.

Table 6 Comparisons with the state-of-the-art methods

Methods	Performance metrics	Leukemia	DLBCL	Lung cancer	Colon cancer	Prostrate cancer	Central nervous system
(Salem et al., 2017) [33]	Accuracy	0.7333	0.8750	0.3784	0.6923	0.5714	0.8462
	Recall	1.0000	1.0000	0.5000	0.6250	0.9412	1.0000
	FPR	0.5000	0.0000	0.2143	0.4286	0.5000	0.0000
	Precision	0.7333	0.8667	0.1304	0.8333	0.5926	0.8000
	F measure	0.8462	0.9286	0.2069	0.7143	0.7273	0.8889
	Entropy	0.0988	0.0539	0.1154	0.0660	0.1347	0.0775
(Ayyad et al., 2019) [38]	Accuracy	0.7907	0.6697	0.8634	0.6086	0.6023	0.8361
	Recall	0.8303	0.8990	0.7647	0.7597	0.7272	0.8013
	FPR	0.2694	0.5383	0.1091	0.5115	0.4556	0.1961
	Precision	0.8582	0.6339	0.4397	0.5562	0.4986	0.4838
	F measure	0.8440	0.7436	0.5584	0.6422	0.5916	0.6033
	Entropy	0.0570	0.1255	0.1569	0.1417	0.1507	0.1526
(Gush et al., 2019) [34]	Accuracy	1.0000	0.9375	1.0000	0.7846	0.9000	0.7538
	Recall	1.0000	0.9833	1.0000	0.7718	0.8926	0.7586
	FPR	0.0000	0.1952	0.0000	0.2352	0.0796	0.1150
	Precision	1.0000	0.9349	1.0000	0.8150	0.9308	0.8893
	F measure	1.0000	0.9585	1.0000	0.7928	0.9113	0.8188
	Entropy	0.0000	0.0273	0.0000	0.0724	0.0290	0.0453
(Rani et al., 2019) [35]	Accuracy	0.9333	1.0000	1.0000	0.8462	0.9375	0.6923
	Recall	0.8889	1.0000	1.0000	1.0000	0.8889	0.5500
	FPR	0.0000	0.0000	0.0000	0.1667	0.0395	0.2444
	Precision	1.0000	1.0000	1.0000	0.3333	0.9194	0.5000
	F measure	0.9412	1.0000	1.0000	0.5000	0.9000	0.5067
	Entropy	0.0000	0.0000	0.0000	0.3662	0.0761	0.3466
Uzma et al. (Proposed work)	Accuracy	0.9000	0.9750	0.9946	0.8462	0.9857	0.8462
	Recall	0.8333	1.0000	1.0000	1.0000	0.9556	0.7500
	FPR	0.0000	0.0364	0.0286	0.1667	0.0000	0.1111
	Precision	1.0000	0.9333	0.9935	0.3333	1.0000	0.7500
	F measure	0.9000	0.9636	0.9967	0.5000	0.9765	0.7500
	Entropy	0.0000	0.0608	0.0063	0.3662	0.0000	0.2158

For this, mainly three combinations of these parameters are made. The first combination is to set Top*N* to 100 with various mutation rates, such as 50, 80, 100, 30, 20, and 10. The second combination is to set Top*N* at 50 and evaluate various mutation rates, such as 30, 20, and 10. The third and the last combination is to set Top*N* as 30 and use 10 and 20 as a mutation rate for all datasets. For each combination, 6, 3, and 2 experiments are conducted respectively using all six datasets as shown in Fig. 6. This experiment represents that the combination 100-80 perform better and give an average accuracy of 92%. In Fig. 6 the combination of 30-30 for parameters *TopN* and the mutation rate shows low accuracy. The value of 30 for *TopN* represents that the ensemble method aggregates the top 30 ranked features by the filter methods. Where, setting 30 as a mutation rate means that 30 genes of the chromosome are swaped during mutation operation. The gene expression dataset contains an enormous number of features. Therefore, for the small values of *TopN* there is a chance of avoiding the relevant features resulting in low accuracy. Results suggest that the largest dataset achieves an accuracy of 98% for the 100-80 setting. However, for this setting recall and FPR values are 95% and 0%, respectively. Values obtained for precision and F-measure are 100% and 97%, respectively. Considering the smallest datasets, the accuracy and recall is 90%, and 83%. Where the value of FPR is 0%, and precision and entropy are 90% and 0% respectively. For the second largest dataset, i.e., lung cancer, values of 99%, 100%, 0.02, 99%, 99%, and 0% are achieved for accuracy, recall, FPR, precision, F-measure, and entropy sequentially. Whereas, for CNS dataset the value for accuracy, recall, FPR, precision, F-measure and entropy are 84%, 75%, 0.11%, 75%, 75%, and 0.215, respectively. The DLBCL dataset gives 97% accuracy and 100%, recall. The FPR value for this dataset is 0.036. The precision, F-measure and entropy values are 93%, 96%, and 0.060. The accuracy for colon cancer on this combination is 84% and for recall its given 100%. However, its PFR, precision, F-measure, and entropy values are 0.1, 33%, 50%, and 0, sequentially. The aforementioned combination yields better results, therefore the proposed framework uses the same setting for the remaining experiments. The average accuracy of all datasets on 100-80 setting, is 94%

4.5 Experiments for using various classifiers

This experiment is carried out to show that the proposed framework is independent of any specific classifier. Therefore, three classifiers are used here, namely, SVM, *k*-NN, and RF. For *k*-NN classifier, various values of *k* are used such as 3, 5, and 7 to identify the best case. For these values of *k*, performance of the proposed work on the *k*-NN classifier is shown in Table 4. Performance of the proposed work on random forest classifier is shown in Table 5.

4.6 Analysis and comparison

The DNA microarray measures the gene expression level of multiple genes simultaneously. The analysis of gene expression datasets is challenging due to its smaller number of samples and a large number of genes. Most of the genes are irrelevant and redundant, therefore, feature selection here plays a vital rule for analysis of such complex data. For feature selection, various techniques are used such as filter, wrapper, and ensemble methods. The present work used the hybrid method that combines the ensemble of three filter methods with a genetic algorithm for feature selection. The novelty of the proposed solution is that it used the unsupervised deep learning method for the evaluation of the chromosomes in a genetic algorithm. The chromosome used here opted for a binary representation where 1 represented the selected feature, and 0 presented the neglected feature. The autoencoder-based *k*-means clustering was used to evaluate the chromosome. The cluster validity index DBI were used as a fitness value for the chromosomes. Once the features were selected, the SVM-based classification was used for the predications of samples. The confusion matrix was used for the evaluation of the proposed framework. The evaluation of the proposed idea on six benchmark datasets was done, namely, leukemia, DLBCL, colon, lung, prostate cancer, and CNS. The proposed approach was evaluated by conducting three types of experiments. These experiments were categorized based on the evaluation of the selected features, setting the parameters Top$N$, mutation rate, and using various classifiers, such as SVM, RF, and KNN with the several value of *k*. The first set of experiments was about the selected feature subset. The feature selection technique was based on an unsupervised approach, therefore the internal and external cluster validity indices were used to evaluate the selected feature sets as shown in Fig. 5.

The second set of experiments conclude that the optimum value of *TopN* is 100 and for mutation rate it is 80. The proposed framework performs better for this combination. The average accuracy on six datasets for the five competing methods is shown in Fig. 9. The method Salem et al. [33] has 57% accuracy on prostate cancer data. Where, for Lung and CNS datasets its accuracy is 37%, and 84%, respectively. It has an accuracy of 87% on DLBCL dataset and on colon cancer data, it has 69% accuracy. For leukemia dataset it obtains 73% accuracy. Salem et al. [33] therefore has an average accuracy of 68% on all datasets. The method Ayyad et al. [38] attains 60%, 86%, 72%, 60%, and 79% accuracies on prostate cancer, lung cancer, DLBCL, CNS, and leukemia dataset, respectively. Whereas, its average accuracy is 72%. The accuracies obtained for Gush et al. [34] for prostate cancer, lung cancer, DLBCL, CNS, and leukemia dataset are 90%, 100%, 89%, 93%, 78%, and 100%, respectively. This algorithm has 89% average accuracy on all datasets. Other than the proposed method, among the four competing methods, better results are obtained for Rani et al. [35] which gives 93% accuracy on the largest and smallest datasets, respectively. For the lung, CNS, and DLBCL datasets, its accuracy is 100%, 90%, and 100%, respectively. However, on the colon cancer data it has 84% accuracy, and the average accuracy on all datasets is 90%. The proposed framework gives an average accuracy of 92%. Individually, on prostate cancer, lung cancer, CNS, DLBCL, colon cancer, and leukemia datasets, its accuracy is 98%, 99%, 92%, 97%, 94%, and 90%. The third experiment used three classifiers, i.e., SVM, *k*-NN, and RF. For the *k*-NN classifier, experiments were also conducted for the various value of *k* on the proposed framework (see Table 4). Using value of *k*=5, better performance of the *k*-NN classifiers is observed. Fig. 7 presents the accuracy of the proposed framework on three classifiers. Fig. 7 lists the accuracy of the proposed work using six datasets and three classifiers. For the prostrate cancer dataset, the accuracy of SVM, *k*-NN, and RF is 98%, 85%, and 94%, respectively. The accuracy for SVM, *k*-NN, and RF is 99%, 80%, and 66% on the lung cancer dataset. Whereas the accuracy on CNS dataset, is 84% for SVM, 69% for *k*-NN, and 97% for RF. They give 97%, 87%, and 77% accuracies on DLBCL dataset. Over the colon dataset an accuracy of 84%, 76%, and 80% is observed. For the smallest dataset leukemia, the accuracy obtained for SVM is 90%, it is 74% for *k*-NN, and 69%, using RF. This suggests that the proposed framework performs better with the SVM classifiers. Therefore, the parameters are adjusted based on these experiments as shown in Table 2. The current proposal is compared with four state-of-art algorithms based on metrics computed using confusion matrix (see Table 6). The performance comparison based on computation time is shown in Fig. 8. While comparing the competing methods based on accuracy, the work in [4] performs better than the other three comparison algorithms [34, 35, 38] with an exception of the central nervous system dataset. Therefore, the proposed work is compared with [35] in terms of accuracy. For two datasets, i.e., prostrate cancer, and central nervous system, the

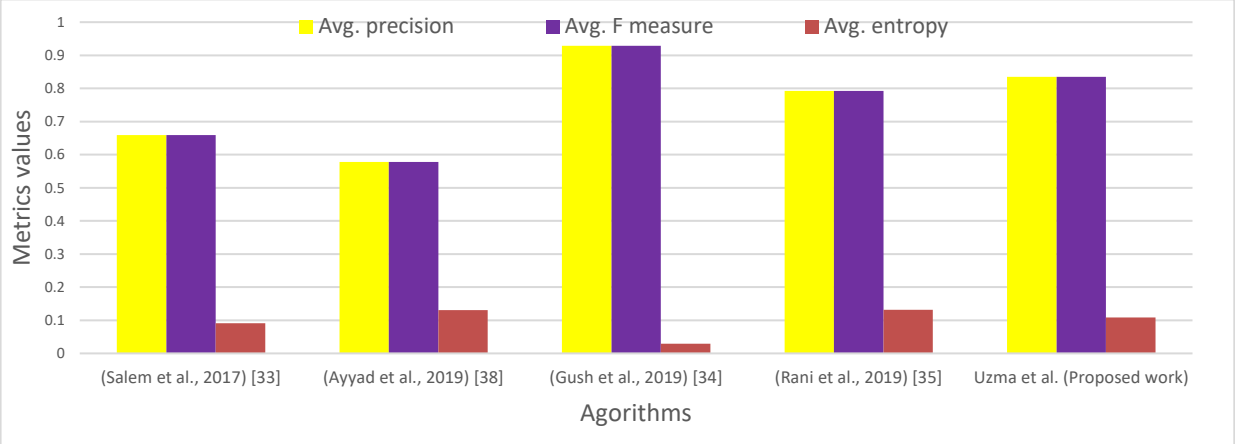

Fig. 10. Average precision, F measures, and entropy of the five competing methods

Table 7 Ranking of the competing approches

Datasets	Uzma et al.	Salem et al. [33]	Ayyad et al. [38]	Gush et al. [34]	Rani et al. [35]
Leukemia	0.5000	0.1667	0.0000	1.0000	0.3333
DLBCL	0.8333	0.3333	0.0000	0.0000	0.5000
Lung cancer	1.0000	0.0000	0.0000	0.1667	1.0000
Colon cancer	0.6667	0.1667	0.0000	0.3333	0.0000
Prostrate cancer	1.0000	0.0000	0.0000	0.0000	0.0000
Central nervous system	0.3333	0.5000	0.0000	0.3333	0.0000
Average	0.7222	0.1944	0.0000	0.3056	0.3056
Std.dev	0.2722	0.1948	0.0000	0.3714	0.4002

Table 8 Paired sample *t*-test

Paired diffrences						T	df	Sig. (2-tailed)
	Mean	Std. deviation	Std. error mean	95% confience interval of the difference				
Pairs				Lower	Upper			
Uzma et al.-Salem et al. [33]	0.528	0.077	0.032	0.461	0.594	2.768	5.000	0.039
Uzma et al-Ayyad et al. [38]	0.722	0.272	0.111	0.488	0.957	11.258	5.000	0.000
Uzma et al-Gush et al. [34]	0.417	-0.099	-0.023	0.466	0.367	2.747	5.000	0.040
Uzma et al-Rani et al. [35]	0.417	-0.128	-0.052	0.527	0.306	2.629	5.000	0.047

proposed work, performs better. However, for four other datasets, it performs the same as others. The average accuracy of all the comparison algorithms is shown in Fig. 9.

While comparing the competing methods based on the recall metric, the proposed idea again performs better for prostrate cancer and central nervous system datasets, however, for other four datasets, it operates the same. Fig. 9 lists the average recall measure for all competing algorithms. Based on false positive rate measure, the proposed idea performs well for all datasets. Fig. 9 shows the average false positive rate of the comparison algorithms. Fig. 10 represent the average precision, F measures, and entropy of the comparison algorithms on all datasets. These measures give better results on prostrate cancer and CNS datasets using proposed framework while perform the same for the other four datasets, i.e., leukemia, DLBCL, lung cancer, and colon cancer. From the comparison table, i.e. Table 6 it is evident that the proposed methodology performs better for all datasets comparatively. The other competing methods did not perform better for the largest and smallest datasets, i.e., prostrate cancer and CNS.

4.7 Statistical significance

The aim of the *t*-test is to show that the selected feature subset is statistically significant. The results of the classification based on the selected features subset are evaluated using the *t*-test here. To show the statistical

significance of the proposed idea, its outcome is compared with four state-of-art algorithms by employing the paired sample *t*-test. First, the null (H_{10}) and an alternative (H_{1A}) hypothesis is defined as follows.

H_{10}: The proposed solution does not perform better based on performance metrics.
$H_{1A,}$: The proposed solution performs better based on performance metrics.

The level of significance α is a probability to reject the null hypothesis which is set to 5%. Where, 95% set as confidence level (1-α) refers to the probability of accepting the null hypothesis. The degree of freedom df presents the total number of datasets, i.e., six. The probability value or *p*-value determine the evidence to reject the null hypothesis. The small *p*-value shows more evidence in favor of an alternative hypothesis.

Based on the six performance metrics, scores are assigned to each approach for all the datasets. The score shows that an approach performs better out of 6 performance metrics for a given dataset. For example, if an approach performs better for 2 performance metrics on a given dataset, the score will be 2 out of 6. Table 7 presents all the assigned scores. Table 8 shows the result of the paired sample *t*-test by using the data mentioned in Table 7. The paired sample *t*-test shows a significant difference between Uzma et al. (proposed) and Salem et al. [t (5) = 2.768241, p < 0.05)], Ayyad et al. [t (5) = 11.25833, p < 0.05)], Gush et al. [t (5) = 2.7466892, p < 0.05)], and Rani et al. [t (5) = 2.6290526, p < 0.05)]. Therefore, this analysis concludes that there is a significant difference between the groups based on the *p*-value. Hence, the null hypothesis (H_{20}) is rejected in favor of the alternative hypothesis.

5 Conclusions and future direction

This work presented a framework that focused on the filter with wrapper-based gene selection for the prediction of cancer. The proposed framework used the ensemble of three filter methods to avoid the chance of not selecting the important genes. This work adopted the meta-heuristic-based algorithm called the genetic algorithm instead of using the conventional wrapper method. This enabled adding or removing the best feature subset from the search space based on the fitness criteria. The deep learning methods have recently been a success in high dimensional data. Therefore, this work adopted the deep learning concept in a meta-heuristic-based wrapper method to efficiently select the best feature subset. The present work used unsupervised feature selection technique for the prediction of cancer. For this, an unsupervised deep learning model called autoencoder was utilized to evaluate the feature subset. The feature subset represented by a chromosome was given as an input to the autoencoder. The present proposal was evaluated on six benchmark datasets using ten standard evaluation metrics. The experiment showed that the clusters generated with the chosen features through the proposed framework were well separated, and the samples in the clusters were associated. Experiments also concluded that the proposed work performs better with the SVM classifier. Additionally, a comparison was made with four state-of-the-art related methods, where the present proposal performed better in most of the cases. This work can be extended in multiple ways in the future. The proposed framework can be extended by selecting the features through a different clustering method, like Self Organizing Maps or density-based clustering. This work utilized unsupervised deep learning approach. One can also extend the current work by selecting features with deep learning in a supervised manner.

Acknowledgement

The authors are indebted to the editor and anonymous reviewers for their helpful comments and suggestions. The authors wish to thank GIK Institute for providing research facilities. This work was sponsored by the GIK Institute graduate research fund under GA-F scheme.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Armstrong, S.A., Staunton, J.E., Silverman, L.B., Pieters, R., den Boer, M.L., Minden, M.D., Sallan, S.E., Lander, E.S., Golub, T.R. and Korsmeyer, S.J (2002) MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nature genetics*, *30*(1):41-47.

2. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A. and Bloomfield, C.D (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, *286*(5439):531-537.
3. Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X. and Powell, J.I (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, *403*(6769):503-511.
4. Jiang, D., Tang, C. and Zhang, A (2004) Cluster analysis for gene expression data: a survey. *IEEE Transactions on knowledge and data engineering*, *16*(11):1370-1386.
5. MacQueen, J (1967) June. Some methods for classification and analysis of multivariate observations. In: Proceedings *of the fifth Berkeley symposium on mathematical statistics and probability.1 (4)*:281-297.
6. Kohonen, T (2012) *Self-organization and associative memory.* Springer, 8
7. Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D (1998) Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, *95*(25):14863-14868.
8. Ben-Dor, A., Shamir, R. and Yakhini, Z (1999) Clustering gene expression patterns. *Journal of computational biology*, *6*(3-4):281-297.
9. Fraley, C. and Raftery, A.E (1998) How many clusters? Which clustering method? Answers via model-based cluster analysis. *The computer journal*, *41*(8):578-588.
10. Brazma, A. and Vilo, J (2000) Gene expression data analysis. *FEBS letters*, *480*(1):17-24.
11. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A. and Bloomfield, C.D (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, *286*(5439):531-537.
12. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A. and Bloomfield, C.D (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, *286*(5439):531-537.
13. Xing, E.P. and Karp, R.M (2001) CLIFF: clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts. *Bioinformatics*, *17*:S306-S315.
14. Law, M.H., Figueiredo, M.A. and Jain, A.K (2004) Simultaneous feature selection and clustering using mixture models. *IEEE transactions on pattern analysis and machine intelligence*, *26*(9):1154-1166.
15. Mahajan, S. and Singh, S (2016) Review on feature selection approaches using gene expression data. *Imp. J. Interdiscip. Res*, *2*(3).
16. Pavithra, D. and Lakshmanan, B (2017) Feature selection and classification in gene expression cancer data. In *2017 International Conference on Computational Intelligence in Data Science (ICCIDS).* IEEE, pp1-6.
17. Alshamlan, H.M., Badr, G.H. and Alohali, Y.A (2015) Genetic Bee Colony (GBC) algorithm: A new gene selection method for microarray cancer classification. *Computational biology and chemistry*, *56*:49-60.
18. Bihani, P. and Patil, S.T (2014) A comparative study of data analysis techniques. *International journal of emerging trends & technology in computer science*, *3*(2):95-101.
19. Halim, Z., Ali, O., & Khan, G (2019) On the Efficient Representation of Datasets as Graphs to Mine Maximal Frequent Itemsets. *IEEE Transactions on Knowledge and Data Engineering*, 1-18.
20. Han, J., Kamber, M. and Tung, A.K (2001) Spatial clustering methods in data mining. *Geographic data mining and knowledge discovery*, pp.188-217.
21. Halim, Z., and Rehan, M (2020) On identification of driving-induced stress using electroencephalogram signals: A framework based on wearable safety-critical scheme and machine learning. *Information Fusion*, 53: 66-79.
22. Gan, G (2013) Application of data clustering and machine learning in variable annuity valuation. *Insurance: Mathematics and Economics*, *53*(3):795-801.
23. Iqbal, S., & Halim, Z (2020) Orienting Conflicted Graph Edges Using Genetic Algorithms to Discover Pathways in Protein-Protein Interaction Networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1-16..
24. Halim, Z., Atif, M., Rashid, A., and Edwin, C.A (2017) Profiling players using real-world datasets: Clustering the data and correlating the results with the big-five personality traits. *IEEE Transactions on Affective Computing*, 10(4):568-584.
25. Oyelade, J., Isewon, I., Oladipupo, F., Aromolaran, O., Uwoghiren, E., Ameh, F., Achas, M. and Adebiyi, E (2016) Clustering algorithms: their application to gene expression data. *Bioinformatics and Biology insights*, *10*: BBI-S38316.
26. Caruana, R. and Freitag, D (1994) Greedy attribute selection. In:*Machine Learning Proceedings*,pp. 28-36
27. Kohavi, R. and John, G.H (1997) Wrappers for feature subset selection. *Artificial intelligence*, *97*(1-2):273-324.
28. Pudil, P., Novovičová, J. and Kittler, J (1994) Floating search methods in feature selection. *Pattern recognition letters*, *15*(11):1119-1125.
29. Frigui, H. and Nasraoui, O (2000) Simultaneous clustering and attribute discrimination. In *Ninth IEEE International Conference on Fuzzy Systems. FUZZ-IEEE 2000 (Cat. No. 00CH37063)*, IEEE. 1:158-163.
30. Chen, H., Zhang, Y. and Gutman, I (2016) A kernel-based clustering method for gene selection with gene expression data. *Journal of biomedical informatics*, *62*:12-20.
31. Song, C., Huang, Y., Liu, F., Wang, Z. and Wang, L (2014) Deep auto-encoder based clustering. *Intelligent Data Analysis*, *18*(6S):S65-S76.
32. Chen, P.Y. and Huang, J.J (2019) A Hybrid Autoencoder Network for Unsupervised Image Clustering. *Algorithms*, *12*(6):122.
33. Salem, H., Attiya, G. and El-Fishawy, N (2017) Classification of human cancer diseases by gene expression profiles. *Applied Soft Computing*, *50*:124-134.

34. Ghosh, M., Adhikary, S., Ghosh, K.K., Sardar, A., Begum, S. and Sarkar, R (2019) Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods. *Medical & biological engineering & computing*, *57*(1):159-176.
35. Rani, M.J. and Devaraj, D (2019) Two-Stage Hybrid Gene Selection Using Mutual Information and Genetic Algorithm for Cancer Data Classification. *Journal of medical systems*, *43*(8):235.
36. Tiwari, S., Singh, B. and Kaur, M (2017) an approach for feature selection using local searching and global optimization techniques. *Neural Computing and Applications*, *28*(10):2915-2930.
37. Langley, P (1994) Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, 184:245-271.
38. Ayyad, S.M., Saleh, A.I. and Labib, L.M (2019) Gene expression cancer classification using modified K-Nearest Neighbors technique. *BioSystems*, *176*:41-51.
39. Muhammad, T., & Halim, Z (2016) Employing artificial neural networks for constructing metadata-based model to automatically select an appropriate data visualization technique. *Applied Soft Computing*, 49:365-384.
40. Shah, A., & Halim, Z (2019) On efficient mining of frequent itemsets from big uncertain databases. *Journal of Grid Computing*, 17(4):831-850.
41. Zhu, X., Li, X., Zhang, S., Ju, C. and Wu, X (2016) Robust joint graph sparse coding for unsupervised spectral feature selection. *IEEE transactions on neural networks and learning systems*, *28*(6):1263-1275.
42. Jiang, P., Maghrebi, M., Crosky, A. and Saydam, S (2017) Unsupervised deep learning for data-driven reliability and risk analysis of engineered systems. In:*Handbook of Neural Computation*, Academic Press, pp. 417-431.
43. Mao, W. and Wang, F (2012) *New advances in intelligence and security informatics*. Academic Press.
44. Breiman, L (2001) Random Forests Machine Learning, 45.
45. Halim, Z. and Khattak, J. H (2019) Density-based clustering of big probabilistic graphs. *Evolving Systems*, 10(3):333-350.
46. Liu, Y., Li, Z., Xiong, H., Gao, X. and Wu, J (2010) Understanding of internal clustering validation measures. In *2010 IEEE International Conference on Data Mining*,IEEE, pp. 911-916.
47. Halim, Z. and Khan, S (2019) A data science-based framework to categorize academic journals. *Scientometrics*, 119(1):393-423.
48. Pakhira, M.K., Bandyopadhyay, S. and Maulik, U (2004) Validity index for crisp and fuzzy clusters. *Pattern recognition*, *37*(3):487-501.
49. Zhu, L., Ma, B. and Zhao, X (2010) Clustering validity analysis based on silhouette coefficient [J]. *Journal of Computer Applications*, *30*(2):139-141.
50. Estévez, P.A., Tesmer, M., Perez, C.A. and Zurada, J.M, (2009) Normalized mutual information feature selection. *IEEE Transactions on neural networks*, *20*(2):189-201.
51. Li, T. and Ma, J (2018) Fuzzy Clustering with Automated Model Selection: Entropy Penalty Approach. In *2018 14th IEEE International Conference on Signal Processing (ICSP)*. IEEE, pp. 571-576.
52. Karypis, M.S.G., Kumar, V. and Steinbach, M (2000) A comparison of document clustering techniques. In:*TextMining Workshop at KDD2000.*
53. Sathiaraj, D., Huang, X. and Chen, J (2019) Predicting climate types for the Continental United States using unsupervised clustering techniques. *Environmetrics*, *30*(4):e2524.
54. Bhuiyan, M.N.Q., Shamsujjoha, M., Ripon, S.H., Proma, F.H. and Khan, F (2019) Transfer Learning and Supervised Classifier Based Prediction Model for Breast Cancer. In *Big Data Analytics for Intelligent Healthcare Management*, Academic Press, pp. 59-86.